AL/CF-SR-1996-0005

ARMSTRONG LABORATORY

HAND GESTURE RECOGNITION USING NEURAL NETWORKS

Paul E. Morton
Edward L. Fix
Gloria L. Calhoun

CREW SYSTEMS DIRECTORATE
HUMAN ENGINEERING DIVISION
Wright-Patterson AFB, OH 45433-7022

MAY 1996

Final Technical Report for Period November 1993 to September 1995

19961002 061

Approved for public release; distribution is unlimited

DTIC QUALITY INSPECTED 3

AIR FORCE MATERIEL COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6753

NOTICES

When US Government drawings, specifications, or other data are used for any purpose other than a definitely related Government procurement operation, the Government thereby incurs no responsibility nor any obligation whatsoever, and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise, as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

Please do not request copies of this report from Armstrong Laboratory. Additional copies may be purchased from:

National Technical Information Service
5285 Port Royal Road
Springfield, Virginia 22161

Federal Government agencies and their contractors registered with Defense Technical Information Center should direct requests for copies of this report to:

Defense Technical Information Center
8725 John J. Kingman Road, Suite 0944
Ft. Belvoir, Virginia 22060-6218

DISCLAIMER

This Special Report is published as received and has not been edited by the Technical Editing Staff of the Armstrong Laboratory.

TECHNICAL REVIEW AND APPROVAL

AL/CF-SR-1996-0005

This report has been reviewed by the Office of Public Affairs (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

The subjects used in this research were under the authority of the Veterans Administration.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

Julie B Cohn

KENNETH R. BOFF, Chief
Human Engineering Division
Armstrong Laboratory

# REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE May 1996	3. REPORT TYPE AND DATES COVERED Final Report, Nov 93 to Sep 95

4. TITLE AND SUBTITLE	5. FUNDING NUMBERS
Hand Gesture Recognition Using Neural Networks (U)	PE 62202F PR 7184 TA 14 WU 6H
6. AUTHOR(S) Paul E. Morton, Col Edward L. Fix, Maj Gloria L. Calhoun	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Armstrong Laboratory, Crew Systems Directorate Human Engineering Division Human Systems Center Air Force Materiel Command Wright-Patterson AFB OH 45433-7022	10. SPONSORING/MONITORING AGENCY REPORT NUMBER AL/CF-SR-1996-0005

11. SUPPLEMENTARY NOTES

The work was supported (in part) by the Department of Veterans Affairs, Rehabilitation Research and Development Center, Edward Hines, Jr. VA Hospital, Hines Illinois 60141, under Agreement Number 93/FR5/166.

12a. DISTRIBUTION/AVAILABILITY STATEMENT	12b. DISTRIBUTION CODE
Approved for public release; distribution is unlimited.	

13. ABSTRACT *(Maximum 200 words)*

Gestural interfaces have the potential of enhancing control operations in numerous applications. For Air Force systems, machine-recognition of whole-hand gestures may be useful as an alternative controller, especially when conventional controls are less accessible. The objective of this effort was to explore the utility of a neural network-based approach to the recognition of whole-hand gestures. Using a fiber-optic instrumented glove, gesture data were collected for a set of static gestures drawn from the manual alphabet used by the deaf. Two types of neural networks (multilayer perceptron and Kohonen self-organizing feature map) were explored. Both showed promise, but the perceptron model was quicker to implement and classification is inherent in the model. The high gesture recognition rates and quick network retraining times found in the present study suggest that a neural network approach to gesture recognition be further evaluated.

14. SUBJECT TERMS	15. NUMBER OF PAGES
gesture recognition, gestural interfaces, input devices, neural networks	34
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UNLIMITED

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

THIS PAGE INTENTIONALLY LEFT BLANK

PREFACE

This report documents an effort conducted in the Human Engineering Division, Crew Systems Directorate of the Armstrong Laboratory, Wright-Patterson Air Force Base, Ohio (7184146H). The work was supported (in part) by the Department of Veterans Affairs (VA), Rehabilitation Research and Development Center, Edward Hines, Jr. VA Hospital, Hines, Illinois 60141, under Agreement No. 93/FR5/166. Specifically, the effort was performed in support of VA Pilot proposal No. C92-453AP ("Recognition of Hand Gestures by People with Motor Impairments: A Feasibility Study") under the National Defense Authorization Act of 1987 which initiated cooperative medical research programs between the VA and Department of Defense (DOD; VHA Directive 10-92-103). The opinions, findings, and recommendations contained herein are those of the authors, and do not necessarily represent those of the VA or DOD.

The authors wish to thank Mr. Ted Morris at the Hines VA Hospital for the advocacy, insight and contribution he provided to this effort. The data analyzed in this effort were collected at the Hines VA Hospital under his direction, following the procedures required by the VA Human Studies Coordination Board.

TABLE OF CONTENTS

INTRODUCTION

Background on Gesture Interfaces

The primary means of nonverbal communication is gestures. There are a variety of static and dynamic signs that have been referred to as "gestures," including: "body language," hand/finger forms, the grasp of open space, involuntary motions, and motions driven by learned customs (Morita, Hashimoto, and Ohteru, 1991). The hand can be considered the primary method by which humans manipulate systems in their environment. Typically, hand manipulations with computer-mediated systems are encumbered with intermediary devices (e.g., key, mouse, and joystick). This report will focus on machine-recognition of hand gestures as an alternative control input to systems. With a *direct* hand-to-machine interface, the operator's hand gesture serves as the sole controller.

Several classifications of controlled operations with hand gestures have been defined. Rhyne (1987) contrasts gestures which point to a specific object/operation with gestures that act to select a set of objects to be involved in an operation. These phases of input, i.e., selecting an object and then selecting an operation, are also performed in mouse pointing interfaces. However, a gestural interface can be designed to accomplish both input phases simultaneously. For instance, the operator can acquire a hand position that has been pre-defined as a grasp operation and the object, over which the hand is positioned, denotes the object to be grasped (Zimmerman, Lanier, Blanchard, Bryson, and Harvill, 1987). Thus, gesture-based control provides an excellent opportunity to consider novel dialogues for the human-machine interface that may be more natural and efficient.

In another gesture classification, Purcell (1985) distinguishes gestural inputs that serve as a command or positioning language separate from gestures that replicate or mimic a task. The latter, often called "scripting by enactment," models gestures as an input to a graphical animation system. Dolan's classification can be viewed as making a similar distinction (Dolan, Friedman, Nagurka, and Gotow, 1987). Occasional gestural inputs to a preprogrammed, semi-autonomous operation are contrasted with real-time dedicated positioning inputs. A taxonomy by Sturman and Zeltzer (1993) is even more detailed; six broad combinations of hand actions and application interpretations are defined. First hand gestures are divided into continuous actions (e.g., moving a finger), and discrete actions (e.g., forming a fist or a waving hand). Next, there are three types of interpretations that can be applied to the hand gesture: 1) direct or master-slave mode of control - hand action maps directly to the task; 2) mapped - hand action is transformed through a continuous mapping to task actions; and 3) symbolic - hand action is interpreted as a symbol which comprises a command to the application.

Besides these classifications, gestural inputs can be categorized by whether they involve specific movements made on a data recording device or involve free limb and hand

movement. In one approach, the operator makes specific gestures, relative to a two-dimensional data tablet, which designate a desired operation. For example, a mouse might be used to pass through displayed points that correspond to symbolic utterances, triggering a synthetic speech system (Horowitz, 1990). Hand-drawn abstract symbols can correspond to specific computer commands (Lipscomb, 1991). These "tablet-based" gesture systems recognize characters based on shape, orientation, size, proportion, velocity and timing of the input signal (Buxton, Flume, Hill, Lee, and Woo, 1983). Syntactical rules are often required to make the system sufficiently robust to differentiate between similar gestures (letter "o" and the number "0", for example). In a second approach, the gesture recognition system learns and recognizes gestures made with free limb and hand movement in three-dimensional (3-D) space and subsequently translates these inputs into specific computer-mediated operations. The remainder of this report will focus on this second approach that involves recognizing a gesture input from the whole-hand in 3-D space.

Gesture Sensing/Measurement

Sturman and Zeltzer (1994), as well as Huang and Pavlovic (1995) provide excellent descriptions of a variety of techniques available to measure the position and angle of body segments and joints used in gestures. Optical, magnetic, and ultrasonic sensing (Zimmerman, et al., 1987) have been used for position tracking. Video-based systems have been demonstrated which involve free-form image-based analysis (Suenaga, Mase, Fukumoto, and Watanabe, 1993; Fukumoto, Suenga, and Mase, 1994; and for American Sign Language recognition, Starner and Pentland, 1995). Non-contact, electric field sensing techniques may enable 3-D position tracking without encumbering gloves and cables (Zimmerman, Smith, Paradiso, Allport, and Gershenfeld, 1995). Movements of a body segment in a dipole field are sensed as changes in displacement current to ground. For measurment of hand and finger joint angles, glove-based techniques are currently the only method that make whole-hand, gesture-based control practical.

A magnetic tracker is typically used with an instrumented glove to provide simultaneous position, orientation and joint-angle data. Magnetic tracking systems use a source element radiating a magnetic field and a small sensor that reports position/orientation with respect to the source. Their accuracy and speed are adequate for real-time gesture measurement. Moreover, this sensing technology does not have a problem with occlusion (e.g., when one finger is in front of another) or maintaining line-of-sight between a sensor and source. Individual fingers, however, are better tracked with glove-based systems.

Glove-based systems are sufficiently lightweight and easily worn so as not to conflict with normal hand activity. These devices are capable of recording and transmitting to a host computer, in real-time, a numeric data-record of an operator's hand/finger shape and dynamics. There are three glove systems that are currently in use

to measure gesture signals: DataGlove™, Dexterous HandMaster™ and CyberGlove™. The construction, sensing system and accuracy of these three systems are compared in McMillan, Eggleston, and Anderson (in press).

- DataGlove™: Developed at VPL Research, Inc. in the late 1980s. Fiber-optic cables run the length of each finger and thumb. Each cable has a light-emitting diode at one end and a phototransistor at the other. Finger flexion bends the cables, attenuating the light they transmit. Light received by the phototransistor is converted into electrical signals proportional to joint angle.

- Dexterous HandMaster™: Exoskeleton-like device worn on the fingers and hand, making it a bit more cumbersome due to increased mass and less stability. Potentiometers at each joint provide highly accurate flexion measurement. System is marketed by EXOS and was developed for use with the Utah/MIT Dexterous Hand Robot. More recent applications include measurements involving fine motor skills and clinical analysis.

- CyberGlove™: Recently introduced by Virtual Technologies, Inc. and is considered state-of-the art. Not only comfortable and easy to use, its accuracy and speed are well suited for complex gestural and fine manipulations. The cloth glove has foil strain gauges sewn into the back; the sensors measure finger and thumb abduction, palm arch, and wrist bending, in addition to finger and thumb joint angles.

Besides these three more sophisticated glove systems, a less accurate measurement of hand position and shape can be obtained with an ultrasonic tracking PowerGlove™ marketed by Mattel in 1989. This flexible molded plastic gauntlet with a Lycra palm is designed to be used as a controller for several Nintendo™ video games.

In some applications of a glove-based system, the operator is also provided with feedback after making a control input/manipulation. This feedback can be provided by vibrotactile, auditory or electrotactile displays (e.g., Massimino and Sheridan, 1993). In one implementation, piezoceramic benders mounted on the glove underneath the finger provide a tingling or numbing sensation to add realism of interacting with virtual objects (Zimmerman, et al., 1987). Without a specific feedback mechanism, operators of gestural interfaces must rely on the system's response to the recognized gesture. For instance, the simulated movement in the direction indicated by a gesture or the synthesized speech response following a sign language gesture provides the operator with feedback on the system's response to a gestural input.

Whole-Hand, Gestural Interface Development

The widespread availability of glove-based recording systems has encouraged their application in a number of computer interfaces. The following describes some research and development efforts along several lines of potential applications.

Natural control interface. Recent advances in developing virtual environments have increased interest in the use of hand gestures as a control interface. One of the first demonstrations of the potential for gestural interfaces was "Put-That-There", a conversational interface for manipulating virtual objects (Bolt, 1980). Users were able to command simple shapes about a large-screen graphics display surface. Hand input devices are also considered intuitive and powerful for control in 3-D environments (Bordegon, 1994). In some implementations, the glove is used in conjunction with a host computer that drives a real-time 3-D computer model of the hand, allowing the glove wearer to "reach" into the surroundings and manipulate computer generated objects as if they were real. This approach was used at NASA/Ames to allow engineers to put their hands into a virtual wind tunnel and allow them to manipulate fluid flow patterns in real-time (Bryson and Levit, 1992). With another system, operators used a Zglove (ultrasonic position/orientation system) to manipulate objects in 3-D (Zimmerman, et al., 1987). Three basic commands were used: grab (fingers closed in a fist), drop (fingers all opened), and copy (few fingers opened). In a system developed by Weimer and Ganapathy at AT&T Bell Laboratories (1989) for experimenting with natural 3-D interfaces, operators wore a DataGlove™ for direct 3-D interaction with the computer models. The model of the hand was built from the thumb and finger data components. In implementing the interface, the index finger tip served as a stylus for locating, and thumb gestures, along with voice commands, were used to initiate a pick. Three gestures were monitored by measuring the abduction sensor on the thumb: picking, moving, or throwing (thumb drawn in towards index finger to select object), clutching (to specify incremental transformation/rotation), and throttling (to scale editing functions where thumb angle scales effect of hand motion). The development of an icon-based notation for describing and documenting gestures was part of an effort to control audio-visual presentation with gestures captured by a DataGlove™ (Baudel and Beaudouin-Lafon, 1993).

It should be noted that many gesture-based control interfaces have included speech recognition in their implementation. In a 3-D modeling system developed by Weimer and Ganapathy (1989), a dramatic improvement in interface utility was realized when speech recognition was added to the gestural commands in the implementation design. The advantages of simultaneous use of spoken commands and gesture inputs was also demonstrated in Bolt's (1980) interface. In another gestural interface (Dolan, et al., 1987), gestures were used to specify "where" and in "what orientation" a robotic action was to be performed and voice commands were used to determine "which" subroutine should be executed. Coupling gestural input with speech recognition can help amplify,

modify, and disambiguate commands from each input modality (see also, Takahashi, Hakata, Shima, and Kobayashi, 1989).

Teleoperation/robotic control. Gestural interfaces also play a key role in virtual environments implemented specifically to control remote systems. Many investigators have explored the possibility of natural and intuitive hand gestures for teleoperation of robots. For instance, gestural interfaces have been used to control dexterous robotic end effectors (Fisher, 1986), a large telerobotic manipulator arm (Hale, 1992); a robot for remote handling in a protected or hazardous factory environment (Mostafa, 1994), and a six-legged mobile robot with manipulator arms (Sturman and Zeltzer, 1993). In the latter application, the investigators examined three different control structures with whole-hand input using a DataGlove™ and conventional input using a set of dials. For low level walking, the whole-hand interface was superior. For high level manipulations, the whole-hand input was on par with the conventional dials. For high-level steering, the whole-hand interface was inferior to conventional dials, because of hand instability and the difficulty exercising control at extreme rotations of the wrist.

Sign language interpretation. Sign language consists of a series of hand gestures and is frequently used to assist communication with nonvocal and/or deaf individuals. Use of a glove-based system during signing may provide sufficient information for automatic recognition of gestures. Besides enabling the signing to serve as a computer input and control, this translation ability can provide a written and/or vocal output of the interpreted message. Machine recognition of gestures made during signing facilitates communication with individuals who do not know or cannot view the visual signs. For instance, a deaf person can "speak" to a hearing person by wearing the TalkingGlove system (Kramer and Leifer, 1989). The CyberGlove™ can convert fingerspelled words from the American Sign Language into synthesized speech for two-way communication. The GloveTalk system developed at the University of Toronto (Fels and Hinton, 1990) also involves mapping hand gestures to a speech synthesizer with a DataGlove™. However, the GloveTalk maps complete hand gestures to whole words, rather than individual letters. The overall hand shape represents a rootword and movement forward and back in one of six directions determines the ending of the root word (each direction coded to a specific ending). The duration and magnitude of the gesture provide data on the rate of speech and stress to be given the word. Obviously, such a system requires more training, but once trained, the communication rate can be faster compared to systems which recognize individual letters. The GloveTalk vocabulary totals 203 words, with 66 root words and 6 endings. The system was not based on an existing sign language and each sign was either static or had limited motion.

Using experienced American Sign Language signers, Quam (1990) examined the basic gesture recognition capabilities of the DataGlove™. In this study, fifteen gestures were reliably recognized with ten flex sensors. A DataGlove™ was also used in an experiment by ATR Research Labs in Japan involving recognition of 46 gestures of the

Japanese kana manual alphabet (Takahashi and Kishino, 1991). A total of 34 out of 46 static gestures were recognized in real-time. The authors noted that hand gestures that are visually different were not always easily distinguished with the DataGlove™. Using the "SLARTI" system, hand gestures involved in Auslan (Australian) sign language were recognized and converted into a format suitable for use by a voice synthesizer (Vamplew and Adams, 1992). The system incorporates position and motion detectors that provide manual components (hand shape, place of articulation, orientation, and movement) of Auslan signs.

Hand measurement research tool. Glove-based systems can also serve as a useful tool in evaluating operator hand function requirements and performance in specialized task environments (Fisher, 1986). For clinical applications, an instrumented glove can provide surgeons and hand therapists with semi-automated, high resolution data for the assessment of initial hand impairment and the evaluation of the results from surgical and/or therapeutic rehabilitation (see Zimmerman, et al., 1987). In a study by Wise, Gardner, Sabelman, Valainis, Wong, Glass, Drace, and Rosen (1990), a glove system was used during a series of range-of-motion tests and found to have application for prosthetic and rehabilitation engineering.

Entertainment. Besides the use of instrumented gloves with computer-based puppetry (Robertson, 1988) and video games, the use of hand gestures in musical performance has probably received the most attention. As early as 1985, Purcell reported that investigators at the Massachusetts Institute of Technology were interested in creating a graphical computer music conductor that combines human body tracker technology and real-time computer music synthesis facilities. In this manner, a "digital" orchestra can perform pre-programmed musical scores under the control of a virtual conductor. A gestural interface developed by Morita et al. (1991) was used to control acoustic parameters in live performances. By instrumenting the conductor's baton, in addition to DataGlove™ measurements, gestures were used to conduct the music. Tracking an infrared light on the baton end with a CCD camera gave tempo information and the position of the baton specified the group of instruments to be played from the electronic orchestra.

Application Considerations for Whole-Hand, Gestural Interfaces

For able-bodied operations, gesture recognition can be used to augment more traditional interfaces (e.g., keyboards or voice input). This alternative control is particularly useful in those workload conditions and operational environments where it is difficult to utilize conventional interfaces. For example, for pilots operating in high noise conditions or experiencing high acceleration, it may be difficult to issue recognizable verbal commands or to reach and select individual control functions. For such environments, it may be useful to have a simple gestural command that will initiate a series of preprogrammed functions until the situation changes such that the pilot can

resume normal operations. Gestural interfaces are also a key control technology proposed for virtual reality applications.

Perhaps the more commonly recognized application of gestural interfaces is in the field of rehabilitation. People with speech limitations and athetoid or spastic movements from stroke or cerebral palsy find interfaces like keyboards, mice or joysticks of limited use. Such people must use "sip-and-puff" controllers, eye-gaze systems, head-mounted joysticks or head-movement control systems. Although these interfaces have some utility, they reduce the freedom of head movement and the number of possible control/command states. Accordingly, better interfaces are needed to extend the independence of people with these limitations.

Thus, gestural interfaces have the potential of enhancing control operations in numerous applications and by both able-bodied and disabled users. Hand gesture recognition may provide a natural, adaptable, and dexterous means for humans to interact with computer systems (Sturman and Zeltzer, 1994). The ability to specify operations with a single intuitive gesture appeals to both novice and experienced operators (Rubine, 1991). Not only can a single gesture be equivalent to many keystrokes and mouse actions, operation of such an interface is silent. Potential disadvantages that need to be considered include the cost, training, communication speed, and accuracy of a gesture recognition system compared to conventional approaches (Rhyne, 1987). Also, transmittal of information with the gesture interface should not conflict with normal hand function (Fisher, 1986). The importance of these factors is dependent on the nature of the task being controlled and the application environment.

Challenges for Whole-Hand, Gestural Interface Design

The variety of plausible applications and the availability of glove-based systems would suggest that gestural interfaces should be in wide use. However, for gestural interfaces to serve as an efficient method of communication, these systems must reliably interpret gestures (Horowitz, 1990). Present gesture recognition systems have difficulty taking into account within and between individual variability. Moreover, these systems have difficulty recognizing the limited and imprecise gestures that are typical of those operating in a less than optimal operational environment or by people with athetoid or spastic movements. The following two steps are key to enabling hand gesture recognition to serve as an effective alternative controller: human factors design and algorithm improvement. Each of these steps is addressed below.

Human factors design. Gestural interface design must take into account the performance of the gesture sensing system and match the human's gestural and manipulation abilities with the coordination and real-time control requirements of the task. Sturman and Zeltzer (1993) provide an excellent "design method for whole-hand input." Their highly disciplined method involves an iterative application of a structured

design flow. First, a series of questions is addressed to determine the feasibility of using whole-hand input for a particular application or set of tasks and whether the gestural interface is natural, adaptable, and dexterous for a particular application. Then, a taxonomy is used to categorize the styles of interaction for whole-hand input. Next, an evaluation guide is applied to decompose the application tasks into specific motions or actions. In this manner, the capabilities of the hand can be compared with task requirements along numerous dimensions (e.g., degrees-of-freedom, hand strength, range-of-motion, speed, steadiness, etc.). Finally, a whole-hand input device is chosen and the interface is tested in an application or simulation. Adherence to a design method such as this will help ensure that application of gestural interfaces will be beneficial to overall system performance.

Algorithm improvement. In that state-of-the-art glove-based systems provide fairly accurate and timely measurements, a second challenge involves improving the algorithms that translate gestural inputs into system commands. General-purpose gesture recognition software typically comes with purchased systems. However, to optimize the speed and accuracy in recognizing the specific set of gestures utilized in a particular interface design, custom algorithm development is recommended. Movement prediction algorithms may also be required for dynamic gestures in order to compensate for system delays. Moreover, emphasis needs to be directed towards improving recognition algorithms such that they are robust to variability within and between individuals and less sensitive to variations induced by less than optimal operational environments (e.g., vibration) and operator hand impairment. For applications involving a continuous stream of gestures, efficient segmentation algorithms are required. Furthermore, the ability to recover from errors and make rapid corrections needs to be programmed. The following section summarizes techniques used to date in processing gesture signals and developing control algorithms.

Gesture Signal Processing

Recognition of static hand gestures is often based on look-up tables that contain minimum/maximum values for each position and joint measurement. More sophisticated algorithms perform some type of pattern analysis on the gesture signal. The data are compared to references established for each hand sensor's degree-of-freedom. To identify a gesture, the match between the data and the reference must be within error tolerances and these tolerances are often weighted by the amount each sensor input contributes to the recognition of the gesture. A variety of statistically based approaches have been utilized, including Bayesian rule-based techniques (Morris, 1994), deformable models (Lanitis, 1995), edge-based techniques (Uras and Veri, 1995), feature analysis (Baudel and Beaudouin-Lafon, 1993), hidden Markov Models (Starner and Pentland, 1995), state-based representation (Wilson and Bobick, 1995), "sum of squares" method (Newby, 1993), and principle component analysis (Takahashi and Kishino, 1991). In the latter reference, both principle component analysis and cluster analysis were used to determine

which fingers, etc. were critical in identifying static hand gestures of the Japanese kana manual alphabet. These analyses provided a rough discrimination among their hand configurations. However, to improve recognition, they established rules for joint bending coding and orientation coding. Incoming gesture signal data were sorted according to these codes and the major principle component. Final matching between a presented hand gesture and the reference hand codes was determined by using the algebraic sum of joint membership values. In their experiment examining real-time gesture recognition, 34 of the 46 hand gestures were recognized correctly.

The above signal processing methods follow more traditional computing techniques by executing instructions in a fixed sequential order. Artificial neural networks offer an alternative approach to signal processing and employ software algorithms which can be trained to learn the relationship that exists between input and output data, including nonlinear relationships (Lippmann, 1987). Not only are neural networks excellent for recognizing patterns in signals, but the algorithms can "learn" from example data and generalize to unseen examples. A neural network is a biologically inspired computational structure composed of many simple, highly interconnected processing elements. These processing elements, or nodes, typically receive signals from several nodes, process this information, and pass a signal onto several more nodes in a manner analogous to biological neurons. The network designer specifies the number of intermediate layers between the input and output units, the number of nodes per layer, as well as the pattern of connections between the layers. Learning is accomplished by adjusting weights, or strength between connections, of the network in order to minimize the performance error over a set of example inputs and outputs. The set of input and output pairs presented to the network during learning is referred to as the training set. Other data sets not used during training are referred to as testing sets.

Using gesture recognition as an application example, conventional processing methods involve a priori determination of what features in the gesture data are important and the development of an algorithm to discriminate these features. With a neural network, the algorithm learns what features are important for distinguishing inputs by comparing gesture inputs with gesture standards in a training set. Moreover, since processing is executed in parallel, use of neural networks increases real-time gesture processing/recognition capability. The result is a system capable of automatically adapting the mapping of an operator's input with the output of the gesture recognizer, tailoring the device control to each individual user or particular operational environment.

In an early application of neural networks for gesture recognition (Kramer and Leifer's Talking Glove, 1989), the algorithm selected the most probable letter from a dictionary of previously stored hand formations that characterized the operator's "gesture signature." During gesture inputs, the dictionary evolved as the recognition algorithm adapted itself to track variations in letter formations. These authors identified a

need to incorporate position and velocity sensors/data to recognize more complex gestures.

In Fels and Hinton's (1990) GloveTalk pilot study, five neural networks were implemented for recognizing hand gestures made with a DataGlove™. Each network's design was tailored to focus on a different aspect of the recognition task: recognizing the root word, word ending, word rate, word stress, and word initiation. For example, the hand shape to root word network used sixteen input nodes (two flex angles per finger and the sines and cosines of the roll, pitch and yaw of the hand). Using a multi-layer perceptron feed-forward network appropriate for nonlinear nodes, a standard back-propagation algorithm was employed. In this manner, the weights assigned to each node were adjusted, in an iterative fashion, until the difference between the desired and actual net outputs was minimized. This study served as a demonstration that neural networks can learn complicated mappings from inputs to outputs; with a 203 gesture-to-word vocabulary, only 7% of the trials resulted in no recognition output and 1% resulted in an erroneous output.

An even more complex task was employed by Vamplew and Adams (1992) in their evaluation of neural network processing for gesture recognition. Simulated CyberGlove™ data for Australian sign language gestures was used in this "SLARTI" pilot study. The processing system was divided into a series of linked smaller sub-networks (20 separate single hidden-layers). Since the temporal components of these signals were not pertinent, a standard feed-forward network using a back propogation algorithm was used for recognizing individual gesture hand shape, location and orientation. For motion and sign classification, though, a time delay neural network topology was employed to utilize the temporal information available in the signals. The hand shape, orientation and location networks served as pre-processors for the motion network which itself served as a pre-processor for the main gesture classification network. Use of multiple networks facilitated independent training, identification of errors and the addition of new gesture signs. After training, the networks were connected by either training additional connection nodes or using standard interactive code (i.e., creating a hybrid system). A "committee system" was also evaluated whereby several nets were trained and presented with the same test data. The output selected by most of the networks was chosen as the system's output. This method was found effective when high levels of noise were present in the signals.

For recognizing a series of individual gestures (i.e., continuous signing), Vamplew and Adams (1992) recommended that post-processing thresholds be added to the network such that a gesture is only recognized if the sign output remains above a magnitude threshold for a certain amount of time determined by a temporal threshold. The individual gesture is also not considered "ended" until the output falls below a different, lower magnitude threshold. Use of two magnitude thresholds would help avoid multiple recognitions of the same gesture, due to noise in the gesture signal. In a later evaluation,

Vamplew and Adams (1995) found that the use of thresholds enabled many sequences to be classified before their actual end, with little impact on accuracy. This "anticipatory classification" leads to the possibility of automatically detecting the individual gestures in a string of continuous commands. A recurrent neural network was also used and found to improve the number and complexity of hand motions that could be recognized.

Neural networks have also been used to interpret dynamic gesture movements recorded with a DataGlove™ for robot control (Brooks, 1989). Multiple Kohonen networks (Kohonen, 1984) operated concurrently on gesture signals to recognize several gestures. Each net was trained to recognize a single gesture, specifically, paths traced by finger motion in *n* - dimensional space of the digit's degrees-of-freedom. Successful recognition of simple gestures was achieved (e.g., closing all the fingers and moving from a neutral hand posture to a grasp position). Brooks concluded that further development was required to realize practical dynamic gesture recognition for robot control.

In a later study, Murakami and Taguchi (1991) used recurrent neural networks to deal with the dynamic processes involved in gestures that specify a word in the Japanese sign language. In a recurrent network, a set of context units provides the system with memory as a trace of processing at the previous time slice. This history is used by the recurrent network to enable recognition of time-series data. In an experiment on the recognition of ten sign language words, the dynamic gesture recognition rate was 96% when a recurrent network was used in conjunction with data encoding/filtering methods.

OBJECTIVE

The objective of this effort was to explore the utility of a neural network-based approach to the recognition of whole-hand gestures. This effort was conducted to assist the Rehabilitation Research and Development Center of the Hines VA Hospital in their effort to recognize hand gestures made by people with athetoid or spastic movement of the forearm or hand. Improvements realized in recognition performance will also benefit the applicability of gestural interfaces as an alternative control for able-bodied operators. In Air Force systems, machine-recognition of hand gestures may facilitate task performance in less than optimal operational environments where use of conventional controls is difficult or impossible.

METHOD

Subject Selection

For neural network development, three right-handed, able-bodied "pilot" subjects were utilized. To validate the neural network approach, ten right-handed "experimental" subjects were utilized: eight subjects had no motor abnormalities and two subjects were stroke patients with hand motor impairments. All subjects were from a research pool

maintained at the Department of Veterans Affairs, Edward Hines, Jr. VA Hospital, Hines Illinois. Subjects were informed of the nature and purpose of the study and were asked to sign consent forms prior to their participation. The data were collected at the Hines VA Hospital and all procedures required by the VA Human Studies Coordination Board were followed.

Materials

A DataGlove™ Model 2, manufactured by VPL Research, Inc., was used to collect gesture related data (Figure 1). This system consists of a glove with 10 fiber-optic joint angle sensors on the thumb and fingers and a Polhemus Fastrak® receiver attached to the back of the glove (top of the hand) with a strong adhesive. The joint angle sensors measured thumb and finger flexure at the inner (metacarpophalangeal) and outer (proximal interphalangeal) joints. The Polhemus component provided six degree-of-freedom location and orientation data (degrees) on the position of the hand. This electronic glove enabled recognition of gestures, regardless of the rotational and lateral position of the hand in 3-D space.

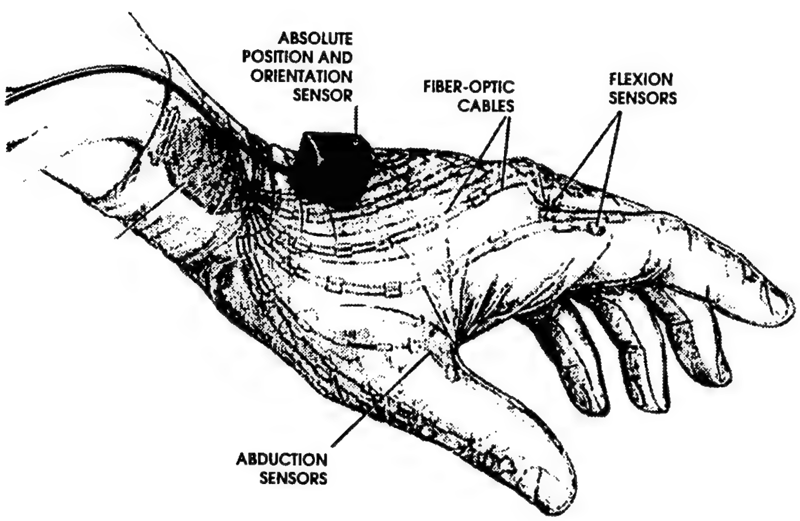


Figure 1. Illustration of the instrumented glove used for gesture data collection.

An 80486 66 MHz PC was used to implement the neural network software.

Procedures

Gesture set. Subjects were asked to perform a subset of the manual alphabet used by the deaf. These gestures were selected because they are static and previous studies have found them to be separable (Quam, 1990). The dynamic letters ("J" and "Z") were excluded as well as characters that are ambiguous or clearly beyond the ability of the DataGlove™ to distinguish (for instance, "R", "U", and "V" are all formed with the index and middle fingers extended, and pointed up.) Appendix A provides an illustration of the 25 gestures examined in this effort. The set includes 22 letters (not "J", "U", "V", and "Z") and the numbers "1", "3", and "5."

Gesture data collection. Subjects were seated at a table and fitted with the DataGlove™ appropriate for the right hand. Standard calibration procedures were conducted according to the system's instructions, with the experimenter assisting the subject in attaining the correct calibration positions.

Next, data collection trials were conducted, with subjects making one gesture per trial. For each trial, the letter to be signed and a pictorial illustration of the corresponding gesture was presented on a computer monitor. Subjects were instructed to adjust their hand/finger positions to mimic the illustrated sign and then push a button with their alternate (left) hand to signify completion of the gesture. Subjects were asked to maintain the gesture for three seconds while multiple data samples were recorded (30 times/second). For each trial, from one to four samples were captured and recorded for further analysis. Thus, there is some variability in the sizes of each individual's data sets for each gesture.

When a new gesture was presented on the monitor, subjects were told to relax their hand for a few seconds and then begin acquiring the next gesture. Subjects were allowed as much time as necessary for relaxing the hand and acquiring gestures, before pushing the button to initiate data collection. For each member of the gesture set, 20 replications were conducted. The presentation order [of the 25 gestures x 20 replications] was random.

Subjects were instructed to notify the experimenter if they knew an error was made in completing the gesture. The experimenter then pressed an "error" key which commanded the data collection system to eliminate the sample recorded and present the same letter command in a later trial.

Neural network design. The multi-layer perceptron network consisted of 12 inputs, 15 hidden nodes, and 25 outputs (see Figure 2). Ten of the 12 inputs were the joint angles, 0-90 degrees, scaled to 0-1 by dividing by 90 (the number of degrees for maximum flexion). The last two were hand orientation direction cosines derived from the quaternion angles. We used the cosine of the angle between the hand's lengthwise (wrist-

to-fingers) axis to the vertical, and the spanwise axis to vertical. The angle between the lengthwise axis and body front-to-back was specifically excluded for two reasons: there are no characters that depend on this angle for recognition, and recognition needs to be invariant to the direction the subject is facing. Appendix B provides an illustration of how the data were transformed and applied.

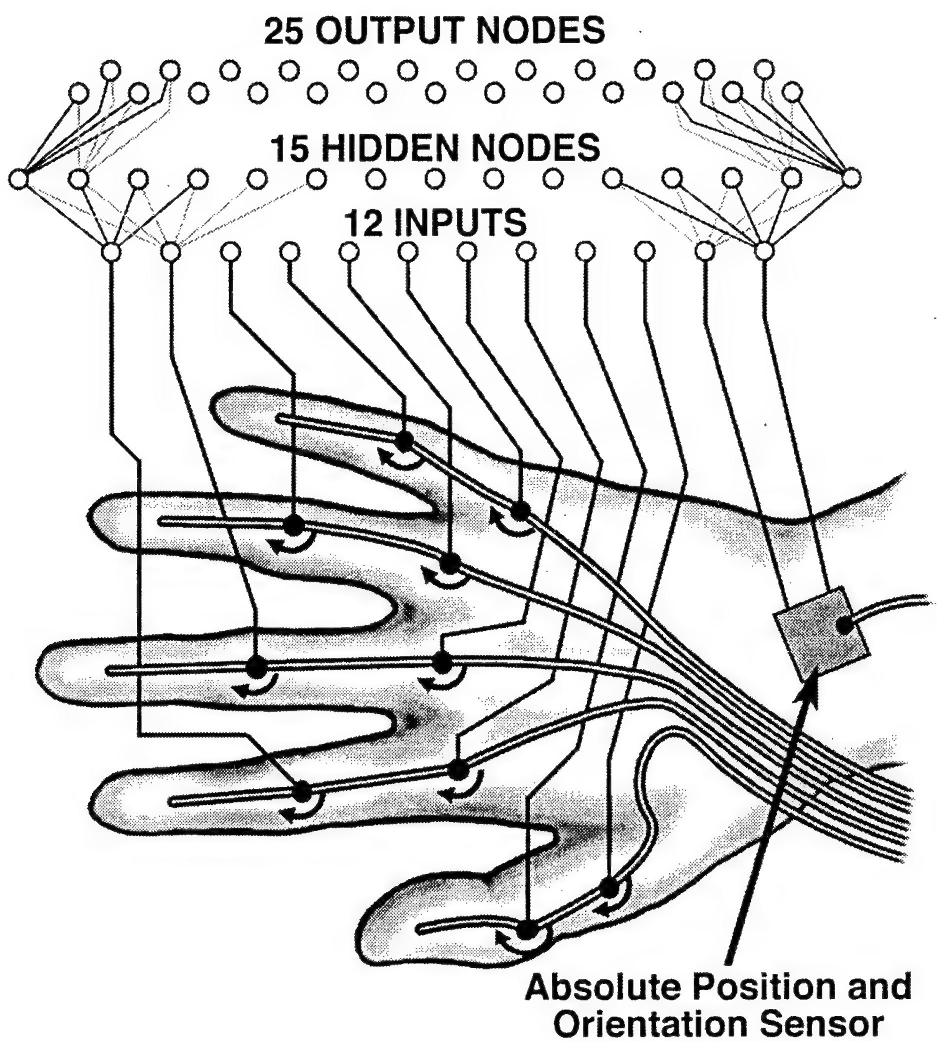


Figure 2. Illustration of the neural network design and source of inputs.

Development and testing of neural network approach for gesture data. Pilot data from three volunteer subjects were used to fine-tune the multi-layer perceptron network and explore alternate network paradigms. Once the network development was finalized using the pilot data, it was applied to the gesture data collected from the ten experimental subjects. Performance of the various implementations was evaluated in terms of percentage total recognition accuracy and the nature of the errors made. The following section provides additional detail on the steps performed and the results found.

RESULTS

Neural Network Development

There were two independent data sets for each of the three pilot subjects and these were first used to examine the effects of training and retraining with the proposed network. Session A sessions were used to train the network and Session B sessions were used to test the network. Figure 3 illustrates the sequence of steps performed and Figure 4 shows the percentage of gestures recognized for each of these manipulations of the perceptron model network.

DATA INPUT Subject	DATA INPUT Session		NETWORK TRAINING SEQUENCE		DATA OUTPUT
1	A	↦	TRAINING		
1, 2, 3	B	↦	TEST	↦	Recognition on Session B
2	A	↦	RETRAINING		
1, 2, 3	B	↦	TEST	↦	Recognition on Session B
3	A	↦	RETRAINING		
1, 2, 3	B	↦	TEST	↦	Recognition on Session B

Figure 3. Illustration of the sequence of steps performed to develop and test the neural network approach for gesture recognition.

The network was first trained on data from Session A of pilot subject 1 (PS1A). This trained network was then tested on Session B data from all three pilot subjects

(PS1B, PS2B, and PS3B). The results are shown in the first bar column of each subject's graph. As to be expected, recognition rates were the highest for PS1 (96.86%), since the network was trained on data from that same subject. However, recognition rates for the two other subjects were still quite good, 76.95% and 66.63% respectively.

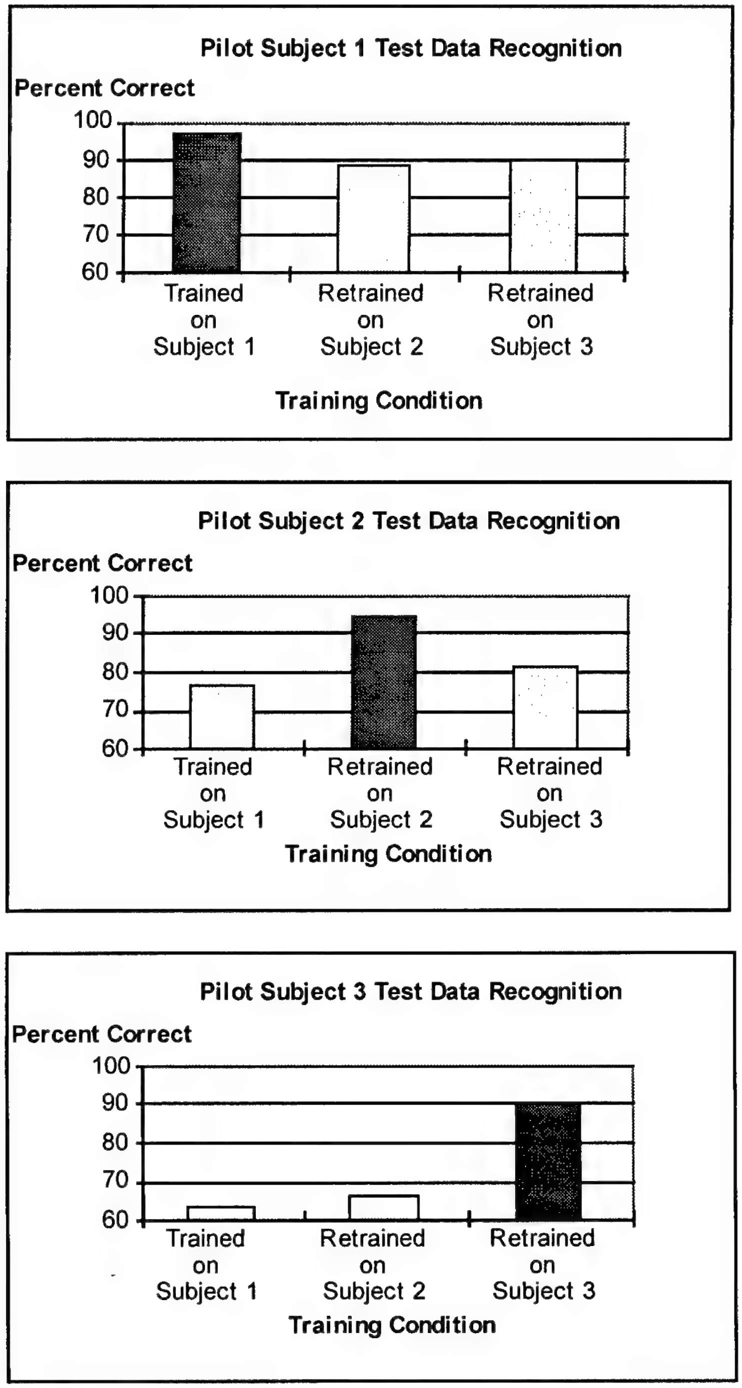


Figure 4. Percent hand gestures recognized for each training/test condition examined with the three pilot subjects.

Next, the network was retrained on data from Pilot Subject 2, Session A. This retrained network was then reapplied to the Session B data from all three subjects. The results are shown in the second column of each graph of Figure 4. Retraining the network increased recognition of PS2's data (94.41%) and left recognition of PS3's data essentially unchanged (65.74%). Recognition of PS1's data dropped by over 8 percentage points to 88.60%.

The final manipulation involved retraining the network on Session A data from PS3 and testing the retrained network on Session B data from the three subjects. The results are shown in the third columns of each graph in Figure 4. Recognition rates increased for both PS1 (slightly, to 90%) and PS3 (dramatically to 90.60%). Recognition performance for PS2 dropped to 81.24%.

A comparison of the results shown in Figure 4 for the three subjects indicates that gesture recognition was very good when the network was trained on the same subject (see shaded columns), averaging 93.95%. While recognition by a trained network on the same subject is quite good, cross speaker recognition suffered. In that it took less than one minute to retrain the neural network, compared to the original network training time of approximately 15 minutes, these results suggest that a trained network can learn a new subject's "gesture style" very quickly and thereafter would perform adequately for that subject.

In a separate procedure, a network was trained with training data pooled from all the pilot subjects. This procedure resulted in recognition rates in the 92-95% level for pilot subject test data.

Neural Network Validation

The network trained on data pooled from all the pilot subjects was tested on data from the ten experimental subjects. These novel subject data were recognized in the 40-65% range. Retraining a base network on an individual is clearly the superior approach.

Therefore, the base network initially trained on one of the pilot subjects (PS1) was retrained on one set of each of the ten experimental subjects. Then, this retrained network was tested on novel data from the same subjects. Table 1 shows the recognition accuracy obtained for each experimental subject, after retraining the base network and reapplying the network.

As can be seen in these data, recognition rates are lower for those data sets that were smaller or had recording problems. Nevertheless, with these subjects, the lowest recognition rate was 79% and that was obtained with a subject with no motor impairments. Averaged recognition rate for the subjects with motor impairments (86.28%) was slightly lower than that for the able-bodied subjects (92.28%). Overall,

recognition using the perceptron neural network model on the data recorded from the DataGlove™ was quite good.

Table 1.
Recognition Rates for Experimental Subject Data with Retrained Neural Network

Subject	Percent	Note
Able-bodied:	**Correct**	
1	79.09	small data set
2	92.91	
3	93.85	
4	99.34	
5	97.89	
6	85.94	recording errors noted
7	89.96	
8	96.40	
Motor Impaired:		
9	82.81	small data set
10	94.48	small data set

The data were also inspected to identify common sources of errors. Table 2 shows the common pairs of gestures in which the subject was trying to form one of the gestures, and the system classified it as another. The gesture pairs are ordered according to how many subjects exhibited the confusion. More than half of the pairs were confused by more than one subject and the common confusions involved 12 members of the gesture set. However, 50% of the confusions were made by two of the ten subjects (Subjects 1 and 7). The other eight subjects had four or fewer pairs of gestures that were confused. This aspect of the data also indicates that gesture recognition with this approach is quite good. For the majority of subjects, there were very few gestures that were confused.

Further examination of the signs for the confused letters suggests that many errors can be attributed to limitations in the DataGlove™ recording system. Any application of a gestural interface would need to address the sensor limitations of the measurement system and either develop hand position sensors to record the required data or develop a gesture vocabulary that matches the capabilities of available sensors.

Table 2.
Letters Commonly Confused

Letters Confused	Subject
A/S	1,2,7,8
1/D	2,4,6,7
A/T	3,6,8
Q/P	1,6,7
O/E	1,3
M/N	1,7
N/T	6,7
1/L	7,8
1/V	1
O/C	1
Q/L	1
S/T	2
D/L	2
S/E	7

Alternative Network Paradigm

In the development of the neural network architecture used in this effort, alternate network paradigms were considered. One feature map network, the Kohonen self-organizing feature map, was also implemented with the pilot subject data to further explore its utility. A 25 by 25 node map architecture was used. Initial training employed a neighborhood size of four nodes in each direction, and inputs were normalized to unit length vectors before comparison and training.

Although the Kohonen network worked well, it required more time to implement and the results were similar to that found with the perceptron architecture. The Kohonen feature maps, though, nicely illustrate how gestures can be confused. For example, Figure 5 illustrates the gestures for "A" and "S" and provides the corresponding Kohonen feature map. The similarity of the feature space available to the networks illustrates the similarity of the gestures themselves and the importance of thumb sensors in the glove-based systems. Thus, Kohonen feature maps can be utilized in the selection of an optimal gesture set.

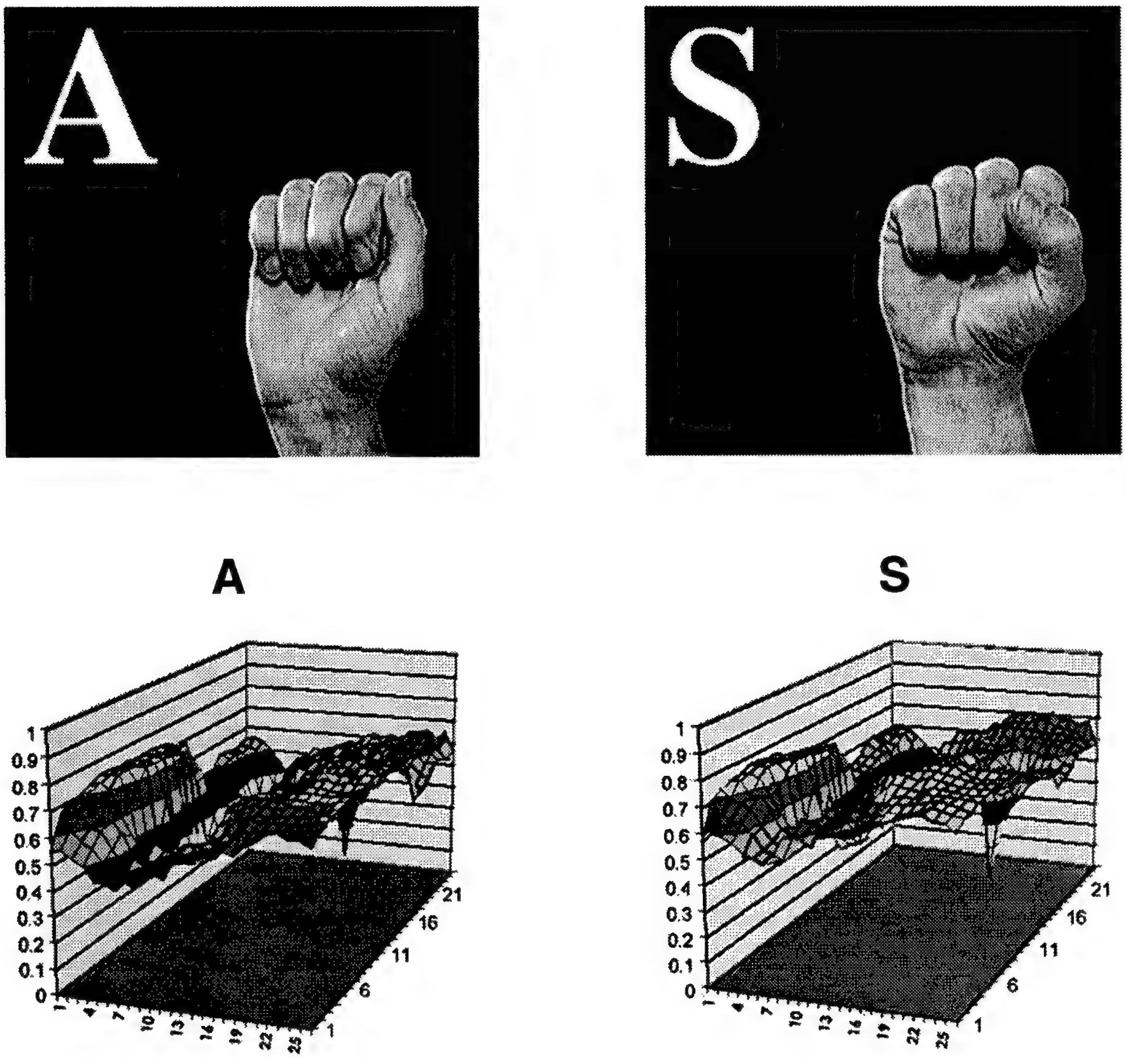


Figure 5. Illustration of "A" and "S" hand gestures and corresponding sample Kohonen feature maps.

CONCLUSIONS

The results of this pilot study provide further evidence that neural networks are very useful in the implementation of gesture recognition systems. Both the multi-layer perceptron neural network and the Kohonen self-organizing feature map were explored. Both showed promise, but the perceptron model was quicker to implement and classification is inherent in the model. For the data collected in the present study, recognition performance was quite good; the system was capable of distinguishing gestures for the majority of subjects. Of special significance is the fact that the system performed adequately for the two subjects with hand motor impairments.

The present pilot study, however, only utilized a small sample size and static hand gestures, one gesture per experimental trial. Further research is required with a larger sample of subjects and an experimental paradigm that directly compares recognition rates obtained with a neural network approach with other candidate approaches. In this manner, the relative payoff of using neural networks can be quantified. Also, further design and investigation are required to develop techniques for recognizing gestures that involve motion and identifying gestures in a string of commands.

The high recognition rates and quick network retraining times found in the present study suggest that, with further development, a neural network approach to gesture recognition will provide algorithms that are sufficiently robust to handle between and within subject variability. Moreover, the "learning" capacity of neural networks should enable the system to be adaptable to signal changes due to fatigue and/or motor impairments or less than optimal operational environments (acceleration, vibration, etc.). It is recommended that these findings be used as an impetus for development of an improved neural network based gesture recognition prototype for further evaluation with able-bodied and disabled subject populations.

REFERENCES

Baudel, T., and Beaudouin-Lafon, M. (1993). Charade: remote control of objects using free-hand gestures. *Communications of the ACM, 36* (7), 28-35.

Bolt, R.A. (1980). "Put-that-there": Voice and gesture at the graphics interface. *Computer Graphics, 14* (3), 262-270.

Bordegon, M. (1994). An environment for the specification and recognition of dynamic gestures. *Journal of Visual Languages and Computing, 4* (3), 205-225.

Brooks, M. (September, 1989). The DataGlove as a man-machine interface for robotics. *2d IARP Workshop on Medical and Healthcare Robotics*, Newcastle upon Tyne, UK, pp. 213-225.

Bryson, S., and Levit, C. (1992). The virtual wind tunnel. *IEEE Computer Graphics and Applications, 12* (4), 25-34.

Buxton, W., Flume, E., Hill, R., Lee, A., and Woo, C. (1983). Continuous hand-gesture driven input. *Proceedings of the Graphic Interface '83*, pp. 191-195.

Dolan, J.M., Friedman, M.B., Nagurka, M.L., and Gotow, J.K. (1987). Gestural control of industrial robots: An application to surgical instrument positioning. *Advanced Topics in Manufacturing Technology: Product Design Bioengineering and Space Commercialization*, Presented at the Winter Annual Meeting of the American Society of Mechanical Engineers, New York, NY: ASME, pp. 29-34.

Fels, S.S., and Hinton, G.E. (1990). Building adaptive interfaces with neural networks: The Glove-Talk Pilot Study. In D. Diaper (Ed.) *Human-Computer Interaction - INTERACT '90*, North-Holland: Elsevier Science Publishers B.V., pp. 683-688.

Fisher, S.S. (1986). Telepresence master glove controller for dexterous robotic end-effectors. *SPIE Vol. 726 Intelligent Robots and Computer Vision: Fifth in a Series*, 396-401.

Fix, E.L., and Rolek, E. (1991). *Modeling a Human Operated Soviet SAM System using Neural Nets* (AL-TR-1991-0016). Wright-Patterson Air Force Base, OH: Armstrong Laboratory.

Fukumoto, M., Suenga, Y., and Mase, K. (1994). Finger-pointer: Pointing interface by image processing. *Computers and Graphics, 18* (5), 633-642.

Hale, J.P. (1992). Anthropometric Teleoperation: Controlling Remote Manipulators with the DataGlove. (NASA-TM-103588) (NTIS:N92-28521/2/XAB).

Horowitz, D.M. (1990). Toward gestural control of synthetic speech for augmentative communication, *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 12* (3), 1315-1317.

Huang, T.S. and Pavlovic, V.I. (1995) Hand gesture modeling, analysis, and synthesis. *International Workshop on Automatic Face- and Gesture-Recognition*, Zurich, pp. 73-79.

Kohonen, T. *Software-Organization and Associative Memory.* Springer, 1984.

Kramer, J., and Leifer, L. (1989). The "Talking Glove": A speaking aid for nonvocal deaf and deaf-blind individuals. *Proceedings of the RESNA 12th Annual Conference*, pp. 471-472.

Lanitis, A. (1995). Automatic interpretation of human faces and hand gestures using flexible models. In M. Bichsel (Ed.) *Proceedings of the International Workshop on Automatic Face- and Gesture- Recognition*, Zurich, pp. 98-103.

Lippmann, R.P. (April, 1987). An introduction to computing with neural nets. *IEEE ASSP Magazine*, pp. 4-22.

Lipscomb, J.S. (1991). A trainable gesture recognizer. *Pattern Recognition, 24* (9), 895-907.

Massimino, M.J., and Sheridan, T.B. (1993). Sensory substitution for force feedback in teleoperation. *Presence, 2* (4), 344-352.

McMillan, G.R., Eggleston, R.G., and Anderson, T.R. (in press). Nonconventional controls. In: G. Salvendy (Ed.), *Handbook of Human Factors and Ergonomics*, Second Edition, New York, NY: John Wiley & Sons.

Morita, H., Hashimoto, S., and Ohteru, S. (1991). A computer music system that follows a human conductor. *Computer, 24* (7), 44-53.

Morris, T. (1994). Personnel communication. IL: Hines VA Hospital.

Mostafa, D.G.M. (1994). Anthropomorphic interface for robot arm programming through a data glove. *Proceedings of the 1994 IEEE International Symposium on Industrial Electronics*, pp. 326-328.

Murakami, K. and Taguchi, H. (1991). Gesture recognition using recurrent neural networks. *Proceedings of the 1991 CHI Conference on Human Factors in Computing Systems*, pp. 237-242.

Newby, G.B. (Summer, 1993). Gesture recognition using statistical similarity. *Proceedings of the Virtual Reality and Persons with Disabilities Conference*, California State University, Northridge.

Purcell, P. (September, 1985). Gestural input to interactive systems. *Computer Bulletin*, Vol. 1, part 3, pp. 3-7.

Quam, D.L. (1990). Gesture recognition with a DataGlove. *Proceedings of the 1990 IEEE National Aerospace and Electronics Conference*, pp. 755-760.

Rhyne, J. (April, 1987). Dialogue management for gestural interfaces. *Computer Graphics, 21* (2), 137-142.

Robertson, B. (July 1988). Mike, the talking head. *Computer Graphics World.*

Rubine, D. (1991). Specifying gestures by example. *Computer Graphics (Proceedings of Siggraph), 25* (4), 239-337.

Rumelhart, D.E., Hinton, G.E., and Williams, R.J. (1986). Learning internal representations by error propagation. In D.E. Rumelhart and J.L. McClelland (Eds.) *Parallel Distributed Processing Volume 1: Foundations*, Cambridge, MA: MIT Press/Bradford, pp. 318-362.

Starner, T., and Pentland, A. (1995). Visual recognition of American Sign Language using Hidden Markov Models. In M. Bichsel (Ed.) *Proceedings of the International Workshop on Automatic Face- and Gesture- Recognition*, pp. 189-194.

Sturman, D.J., and Zeltzer, D. (1993). Utility of whole-hand input. *SPIE Vol. 2057*, pp. 282-291.

Sturman, D.J., and Zeltzer, D. (1994). A survey of glove-based input. *IEEE Computer Graphics and Applications*, *23*, 30-39.

Suenaga, Y., Mase, K., Fukumoto, M., and Watanabe, Y. (1993). Human reader: An advanced man-machine interface based on human images and speech. *Systems and Computers in Japan*, *24* (2), 88-102.

Takahashi, T., Hakata, A., Shima, N., and Kobayashi, Y. (1989). Unifying voice and hand indication of spatial layout. *SPIE Volume 1198 Sensor Fusion II: Human and Machine Strategies*, pp. 346-353.

Takahashi, T., and Kishino, F. (April, 1991). Hand gesture coding based on experiments using a hand gesture interface device. *SIGCHI Bulletin, 23* (2), pp. 67-74.

Uras, C., and Veri, A. (1995). Hand gesture recognition from edge maps. In M. Bichsel (Ed.) *Proceedings of the International Workshop on Automatic Face- and Gesture-Recognition*, Zurich, pp. 116-121.

Vamplew, P., and Adams, A. (March, 1992). The SLARTI System: Applying artificial neural networks to sign language recognition. *Proceedings of the Technology and Persons with Disabilities Conference*, California State University, Northridge, pp. 575-579.

Vamplew, P., and Adams, A. (1995). Recognition and anticipation of hand motions using a recurrent neural network. *Proceedings of the IEEE International Conference on Neural Networks*, Perth, Western Australia.

Weimer, D., and Ganapathy, S.K. (1989). A synthetic visual environment with hand gesturing and voice input. *Proceedings of the CHI.* New York, NY: ACM Press, 235-240.

Wilson, A.D., and Bobick, A.F. (1995). Configuration states for the representation and recognition of gestures. In M. Bichsel (Ed.) *Proceedings of the International Workshop on Automatic Face- and Gesture- Recognition*, Zurich, pp. 129-134.

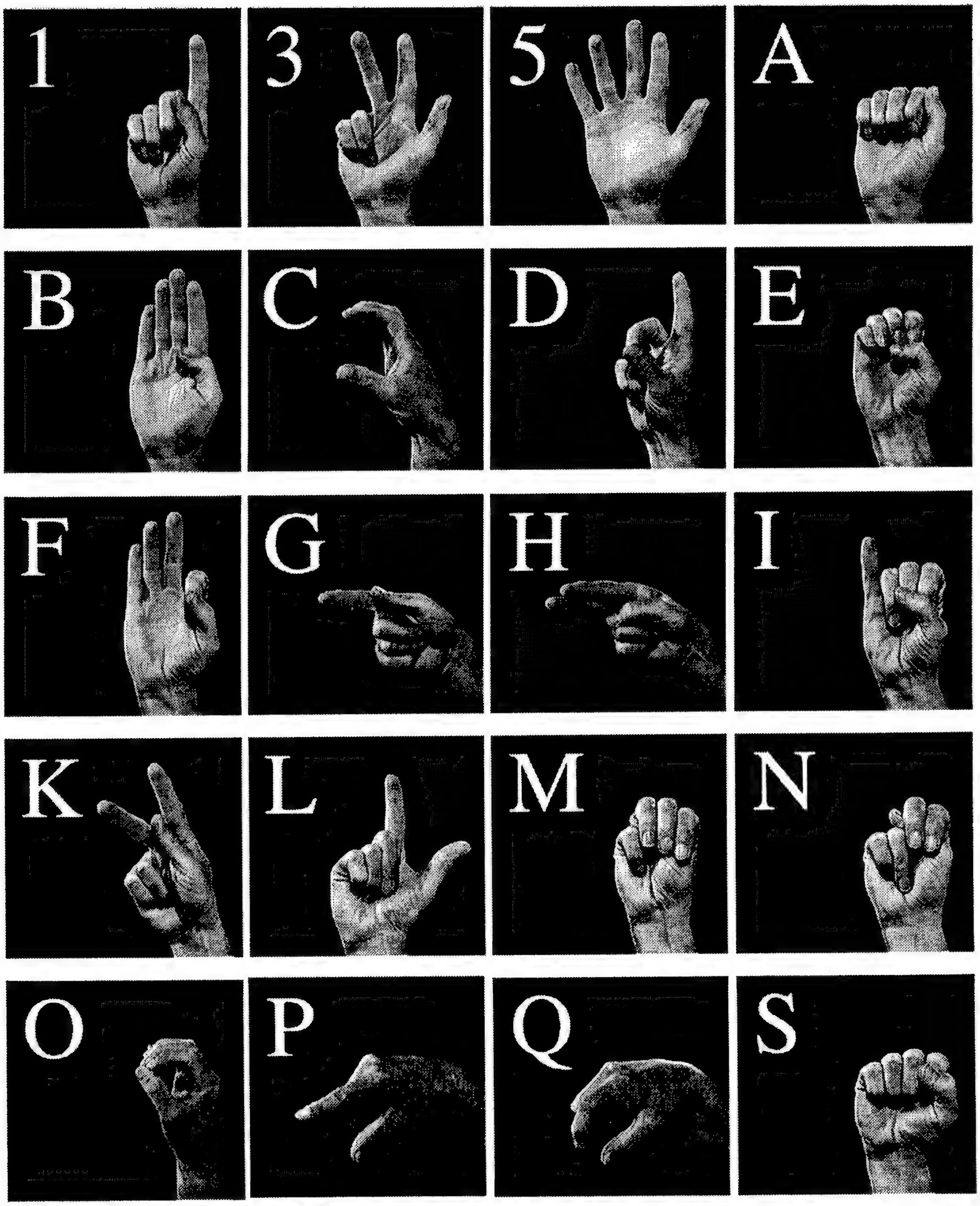
Wise S., Gardner, W., Sabelman, E., Valainis, E., Wong, Y., Glass, K., Drace, J. and Rosen, J. (1990). Evaluation of a fiber optic glove for semi-automated goniometric measurements. *Journal of Rehabilitation Research and Development*, *27* (4), 411-424.

Zimmerman, T.G., Lanier, J., Blanchard, C., Bryson, S., and Harvill, Y. (1987). A hand gesture interface device. *Proceedings of the Human Factors in Computing Systems and Graphics Interface,* New York, NY: ACM Inc., pp. 189-192.

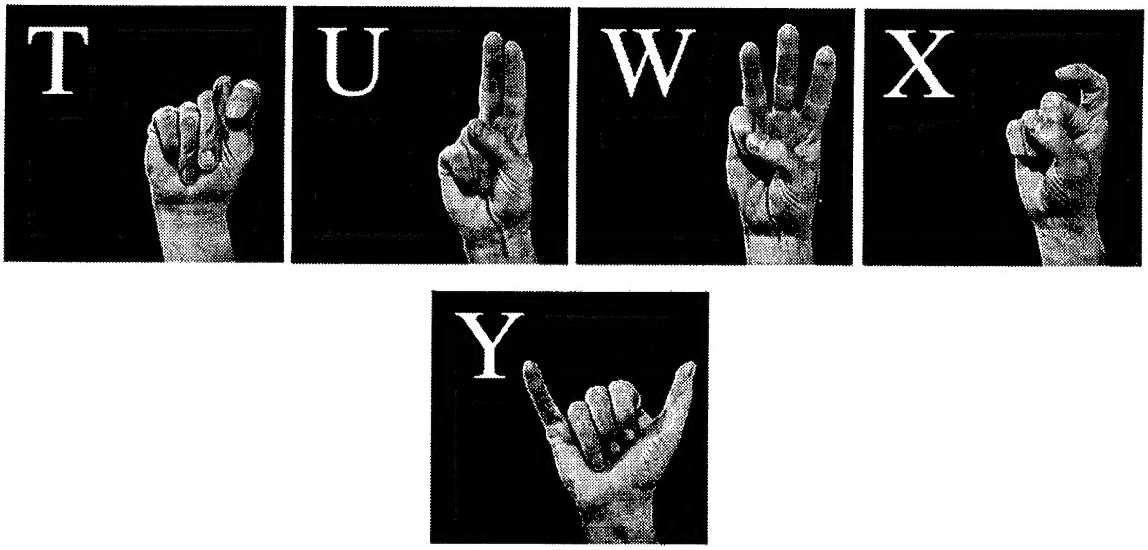
Zimmerman, T.G., Smith, J.R., Paradiso, D.A., Allport, D., and Gerschenfeld, N. (1995). *Proceedings of the CHI.* New York, NY: ACM Press.

Appendix A

Gestures Used for Recognition



Gestures Used for Recognition



Appendix B

Multi-Layer Feed-Forward Neural Networks

Neural networks "... attempt to achieve good performance via dense interconnection of simple computational elements. In this respect, artificial neural net structure is based on our present understanding of biological nervous systems" (Lippmann, 1987).

Single Node Perceptrons

A single computational element or neuromime is shown in Figure 1. The output value is given by

$$y = f\left(\sum_{i=1}^{N} w_i x_i - \theta\right) \tag{B.1}$$

where

$$f(\alpha) = \frac{1}{1+e^{-\alpha}} \tag{B.2}$$

is the sigmoid equation (Figure 2) and x represents an input vector element, w represents the connection weight, and θ is a small random threshold. N is the number of elements in the input vector. It can be shown (Lippmann, 1987) that Equation B.1 describes a hyperplane boundary (a straight line if there are two inputs) in N-dimensional space between two regions. If vectors $\mathbf{x} = \{x_1,...,x_N\}$ which are separable into two regions are applied to the inputs, the weights can be adapted so that the hyperplane divides the two regions of points. The training algorithm, called the delta rule, is

$$\Delta w_i = \eta(d - y)x_i \tag{B.3}$$

$$1 \le i \le N$$

$$0 < \eta < 1$$

where d is the desired output (0 or 1). After a number of training trials, the perceptron may converge to a solution. In this way, the perceptron can classify the input vectors. The output can also be trained to intermediate values between 0 and 1, to approximate continuous functions.

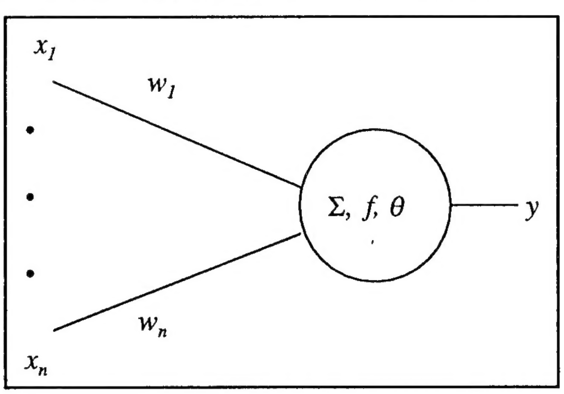


Figure 1. Single Perceptron Node

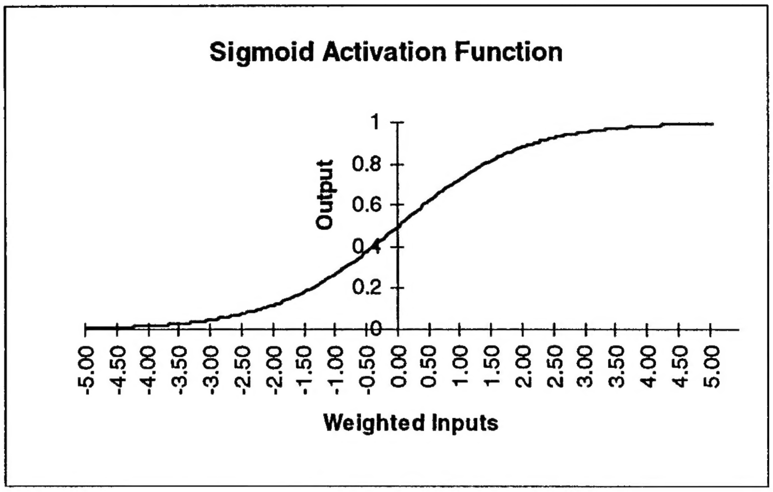


Figure 2. Sigmoid Function

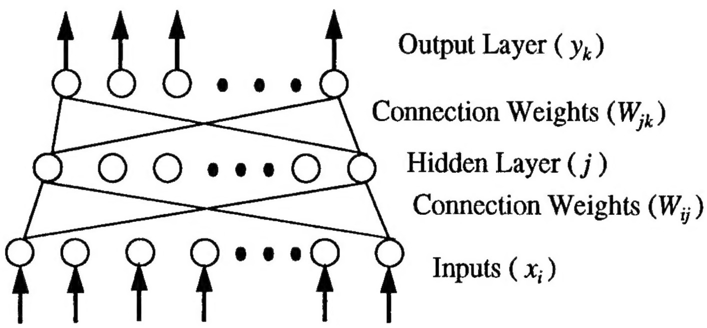


Figure 3. Multi-Layer Feed-Forward Network

Multi-Layer Perceptrons

It can be shown (Lippmann, 1987) that an arrangement of several nodes in each of three layers, where all nodes in one layer (or all inputs) are connected to all nodes of the next layer, can separate an arbitrary number of classes and regions with arbitrarily complex boundaries. This arrangement is schematically shown in Figure 3. The complexity that a network can handle depends on the number of nodes in each layer.

The extended training algorithm is called back propagation, and uses the generalized delta rule:

$$\Delta w_{bc} = \eta \delta_c y_b \tag{B.4}$$

where

$$\delta_c = y_c(1-y_c)(d_c - y_c) \tag{B.5}$$

if the current layer is the output where d_c is the desired output of node c and y_c is the actual output or

$$\delta_c = y_c(1-y_c)\sum_a \delta_a w_{ca} \tag{B.6}$$

if the current layer is an inner or hidden layer. In Equations B.4 through B.6, x denotes an input to a node and y is its output. Note that the output of one node is an input to another in the next layer. The subscript c denotes the current layer, while a denotes the layer above and b denotes the layer below. The θ values in Equation B.1 are also adapted by back propagation. A more complete description and derivation can be found in Rumelhart, Hinton and Williams (1986).

Perceptron Simulation

Although perceptrons are conceptually implemented as massively parallel networks of simple processors, they can be simulated on a conventional digital computer. These simulations are very computation intensive, but if the net is small enough, it may be possible to run the simulation in real time as a subroutine or on an appropriate external processor. The back propagation training algorithm is the most time consuming part, but once the net is trained the weights can be transferred to a real time processor.

This is only one of many different neural network architectures. Others include the Kohonen self-organizing map, the Grossberg ART networks, the Hopfield network, bidirectional associative memory, and many more. Each has its own strengths and potential applications, but detailed descriptions of them would be beyond the scope of this work.

U.S. Government Printing Office 1996 750-071/40011